We claim:

1. A method for displaying computer system runtime information, comprising the steps of:

displaying a plurality of runtime information items in a first hyperbolic tree as a plurality of nodes;

showing one or more links between nodes in said first hyperbolic tree, with said one or more links representing node causal relationships; and

moving a particular tree node of said first hyperbolic tree to a center node position in said first hyperbolic tree if a user input selects said particular tree node.

2. The method of claim 1, wherein the displaying step dynamically generates said first hyperbolic tree.

3. The method of claim 1, wherein said first hyperbolic tree further displays causal relationship runtime information in a non-tree graph.

4. The method of claim 1, wherein a user input to a displayed hyperbolic tree is capable of expanding or contracting said displayed hyperbolic tree.

5. The method of claim 1, further comprising the step of generating a linking graph that links said first hyperbolic tree and a second hyperbolic tree.

6. The method of claim 1, further comprising the step of generating a linking graph that links said first hyperbolic tree and a second hyperbolic tree if a user input is a navigation input that selects said second hyperbolic tree.

7. The method of claim 5, wherein the generating step generates a linking graph that connects a current node of said first hyperbolic tree to a corresponding node in said second hyperbolic tree.

8. The method of claim 5, wherein the step of generating said linking graph further comprises dynamically generating said linking graph.

9. The method of claim 5, further comprising the step of moving into focus a selected node from a second hyperbolic tree, wherein said selected node was in a contracted subtree.

10. A method for displaying computer system runtime information, comprising the steps of:

displaying a plurality of runtime information items in a first hyperbolic tree as a plurality of nodes;

showing one or more links between nodes in said first hyperbolic tree, with said one or more links representing node causal relationships;

moving a particular tree node of said first hyperbolic tree to a center node position in said first hyperbolic tree if a user input selects said particular tree node; and

generating a linking graph that links said first hyperbolic tree and a second hyperbolic tree if a user input is a navigation input that selects said second hyperbolic tree.

11. The method of claim 10, wherein the displaying step dynamically generates said first hyperbolic tree.

12. The method of claim 10, wherein a hyperbolic tree further displays causal relationship runtime information in a non-tree graph.

13. The method of claim 10, wherein a user input to a displayed hyperbolic tree is capable of expanding or contracting said displayed hyperbolic tree.

14. A visualization system adapted for displaying runtime information from a computer system, comprising:

a repository for storing a plurality of runtime information items from said computer system;

a display device capable of displaying one or more runtime information items of said plurality of runtime information items; and

an analyzer for retrieving said one or more runtime information items from said plurality of runtime information items, processing said one or more runtime information items, and generating a display of said one or more runtime information items on said display device;

wherein said display device displays said runtime information as at least two hyperbolic trees.

15. The system of claim 14, wherein said display device communicates with said analyzer over a second communication link.

16. The system of claim 14, wherein said repository communicates with said analyzer over a first communication link.

17. The system of claim 14, wherein said at least two hyperbolic trees comprises an interface repository hyperbolic tree.

18. The system of claim 14, wherein a hyperbolic tree of said at least two hyperbolic trees comprises a dynamic call graph hyperbolic tree.

19. The system of claim 14, wherein said at least two hyperbolic trees represent different aspects of a system characterization.

20. The system of claim 14, wherein said analyzer assigns a unique identifier for each displayed hyperbolic tree node, and wherein said unique identifier is used for cross-linking between tree nodes in a hyperbolic tree.

21. The system of claim 14, wherein said analyzer assigns a unique identifier for each displayed hyperbolic tree node, and wherein said unique identifier is used for cross-linking between a first tree node in a first hyperbolic tree and a second tree node in a second hyperbolic tree.

22. The system of claim 14, wherein said analyzer assigns a unique identifier for each displayed hyperbolic tree node, and wherein said unique identifier is used by said at least one display device to query said analyzer for further runtime information items for a current hyperbolic tree node.

23. The system of claim 14, wherein said display moves into focus a selected node from a second hyperbolic tree, wherein said selected node was in a contracted subtree.

24. The system of claim 14, wherein a tree-specific, node-oriented menu is provided for the user to inspect a node information, a path information, and a subgraph-associated system information.

25. The system of claim 14, wherein said display device is capable of being launched and operated inside a web browser and wherein said display device interacts directly with said analyzer or through a web server.